

Amendment for discussion purposes only. Please DO NOT  
enter! -- /ANB/ 4/21/09

## FAX COVER SHEET

---

TO

---

COMPANY

---

FAX NUMBER 15712733908

---

FROM Lee & Hayes Seattle Office

---

DATE 2009-04-20 22:18:10 GMT

---

RE 10/618,335 Proposed Interview Agenda ( MS1-1538US )

---

## COVER MESSAGE

---

Dear Examiner Boutah,

Please find attached the proposed agenda for the telephonic interview scheduled for April 22, 2009 at 2:00 pm EST.

Thank you and regards,

Pam Prellwitz

Paralegal

206.876.6011

pam<mailto:pam@leehayes.com>@leehayes.com<mailto:pam@leehayes.com>

Lee & Hayes pllc, Intellectual Property Law

701 Pike St, Suite 1600, Seattle, WA 98101 | 206.315.4004 fax |

www.leehayes.com<http://www.leehayes.com>

NOTE: This email and any attachments contain information from the law firm of Lee & Hayes, pllc, that is confidential and/or subject to attorney-client privilege. If you are not the intended recipient of this message, please do not read it or disclose it to others. Instead, please delete it and notify the sender immediately.

Application Serial Number: 10/618,335

Attorney Docket Number: MS1-1538US

To: Examiner Boutah  
Fax: (571) 273-3908  
Phone: (571) 272-3908

From: Brett J. Schlameus (Reg. No. 60827)  
Lee & Hayes, PLLC  
brett@leehayes.com  
(Tel. 206-876-6022; Fax 509-323-8979)

**Agenda and Request for an Examiner Interview**

– INFORMAL COMMUNICATION – FOR DISCUSSION PURPOSES ONLY –

**[0001]** As requested, please find herein an agenda for the interview scheduled for 4/22/2009 at 2:00 pm EST. Thank you for agreeing to discuss this matter.

**Interview Agenda:**

- Discussion of current § 103(a) rejections
- Discussion of exemplary differences between the application/claims and the cited references
- Discussion of proposed amendments

**[0002]** Initially, Applicant thanks the Examiner for granting an interview for the above-referenced case. During the interview, Applicant wishes to discuss the following proposed amendments to independent Claims 1, 11, 18, 26, 27, 32, 37, 39, and 47. Applicant respectfully submits that Claims 1, 11, 18, 26, 27, 32, 37, 39, and 47, as amended as proposed, are allowable over at least the references of record. Please note that the proposed amendments are for discussion purposes only and should not be entered into the official record. I look forward to discussing this with you.

Respectfully Submitted,

Lee & Hayes, PLLC  
Representative for Applicant

Brett J. Schlameus (brett@leehayes.com; 206-876-6022)  
Registration No. 60827

Assistant: Pam M. Prellwitz  
206-876-6011  
pam@leehayes.com

**Appendix of Claims with Proposed Amendments**

1. (Currently Amended) A method, comprising:

requesting data to be streamed from a source device to a client device over a network;

building, by a parent Distributed Media Session ("DMS"), a distributed software infrastructure from an optimized distributed topology, the built distributed software infrastructure configured to stream data to the client device from the source device without rendering the data by the source device, the parent DMS providing a federated mechanism for control; [[and]]

resolving, by the parent DMS, a distributed topology from the request, wherein:  
the distributed topology references a plurality of software components that, when executed, fulfill the request; [[and]]

at least one of the plurality of software components is executable on each of:

the source device; and

the client device[[. ]]

instantiating, by the parent DMS, one or more child DMS; and

delegating control of one or more of the plurality of software components of the distributed software infrastructure to the one or more child DMS.

11. (Currently Amended) A method, comprising:

receiving a request to stream data from a source device to a client device over a network; and

resolving, by a parent Distributed Media Session ("DMS"), a distributed topology

that references software components to fulfill the request, the parent DMS providing a federated mechanism for control, wherein the distributed topology is resolved from:

capabilities of the client device to render a stream of data; and

capabilities of the source device to stream data that is to be rendered; [[and]]

building, by the parent DMS, a distributed software infrastructure from an optimized distributed topology, the built distributed software infrastructure configured to stream data to the client device from the source device without rendering the data by the source device, and building from the distributed topology a distributed software infrastructure that includes the referenced software components, wherein at least one of the software components is executable on each of:

the source device; and

the client device[[.]], and

instantiating, by the parent DMS, one or more child DMS; and

delegating control of the referenced software components included within the distributed software infrastructure to the one or more child DMS.

18. (Currently Amended) A method, comprising:

discovering the capabilities of a client device to render a stream of data;

discovering the capabilities of a source device to stream data that is to be rendered, the discovering the capabilities of the client device and the source device by querying a look-up table that specifies a particular capability;

building, by a parent Distributed Media Session("DMS"), a distributed software

infrastructure from an optimized distributed topology, the built distributed software infrastructure configured to stream data to the client device from the source device without rendering the data by the source device, the parent DMS providing a federated mechanism for control; [[and]]

deriving, by the parent DMS, a distributed topology from both said capabilities the capabilities of the client device and the capabilities of the source device, wherein:

the distributed topology references a plurality of software components to fulfill the request; and

at least one of the software components referenced by the distributed topology is executable on each of:

the source device; and

the client device[[.]] and

instantiating, by the parent DMS, one or more child DMS; and

delegating control of one or more of the referenced plurality of software components to the one or more child DMS.

26. (Currently Amended) A method, comprising:

receiving a request to stream data from a source device to a client device;

discovering the capabilities of the client device to render a stream of data;

discovering the capabilities of the source device to stream data that is to be rendered, the discovering the capabilities of the client device and the source device by querying a look-up table that specifies a particular capability;

deriving, by a parent Distributed Media Session ("DMS"), a distributed topology to

fulfill the request from both said the capabilities of the client device and the capabilities of the source device, wherein the distributed topology references a plurality of software components, the parent DMS providing a federated mechanism for control;

building, by the parent DMS, from the distributed topology a distributed software infrastructure, wherein the distributed software infrastructure includes ~~[[said]]~~ the software components referenced by the distributed topology;

building, by the parent DMS, a distributed software infrastructure from an optimized distributed topology, the built distributed software infrastructure configured to stream data to the client device from the source device to the client device without rendering the data by the source device;

instantiating, by the parent DMS, one or more child DMS;

delegating control of one or more of the plurality of software components referenced by the distributed topology to the one or more child DMS;

streaming the data from the source device to the client device; and

rendering the data by the client device.

27. (Currently Amended) A parent distributed media session ("DMS"), comprising

a software component residing on a storage medium having instructions that, when executed, directs acts comprising:

resolving, by the DMS that provides a federated mechanism for control, a distributed topology that references a plurality of software components that, when executed, fulfill a request to stream data from a source device to a client device;

wherein the resolving further comprises optimizing the distributed topology such that the distributed software infrastructure which is built from the distributed topology is configured to stream data from the source device to the client device without rendering the data by the source device before the data is streamed; [[and]]

building, from the distributed topology, a distributed software infrastructure that includes[[said]] the software components, wherein at least one of the [[said]] software components is executable on each of:

the source device; and

the client device[[.]].

instantiating, by the parent DMS, one or more child DMS; and

delegating control of one or more of the plurality of software components to the one or more child DMS.

32. (Currently Amended) A computer-readable medium comprising computer-executable instructions residing on a storage medium that, when executed, direct a computing device to perform acts comprising:

resolving, by a parent Distributed Media Session ("DMS") and without user intervention, a distributed topology that references a plurality of software components that, when executed, stream data from a source device to a client device over a network;

building, by the parent DMS, a distributed software infrastructure from an optimized distributed topology, the built distributed software infrastructure configured to stream data to the client device from the source device to the client device without



rendering the data by the source device; [[and]] wherein at least one of the plurality of software components is executable on each of:

the source device; and

the client device[[. ]].

instantiating, by the parent DMS, one or more child DMS; and  
delegating control of one or more of the plurality of software components to the  
one or more child DMS.

37. (Currently Amended) A computer-readable storage medium comprising computer-executable instructions that, when executed, direct a computing device to perform acts comprising:

discovering the capabilities of a client device to render a stream of data;

discovering the capabilities of a source device to stream data that is to be rendered, the discovering the capabilities of the client device and the source device by  
querying a look-up table that specifies a particular capability; [[and]]

deriving, by a parent Distributed Media Session ("DMS") and without user  
intervention, a distributed topology from both [[said]] the capabilities of the client device  
and the capabilities of the source device, wherein:

the distributed topology references a plurality of software components that, when executed, stream data, without rendering the data, from the source device to the client device; and

at least one of the plurality of software components referenced by the distributed topology is executable on each of:

the source device; and  
the client device[[. ]],  
instantiating, by the parent DMS, one or more child DMS; and  
delegating control of one or more of the plurality of software components to the  
one or more child DMS.

39. (Currently Amended) A system, comprising:  
a source device that is operable to stream data to be rendered;  
a client device that is operable to render a stream of data; and  
a parent distributed media session ("DMS"), which when executed, causes  
actions to be performed including:

resolving, by the parent DMS, a distributed topology that references a plurality of  
software components that, when executed, stream data from the source device to the  
client device over a network; [[and]]

building, by the parent DMS, from the distributed topology a distributed software  
infrastructure that includes [[said]] the software components, wherein the building  
further comprises building the distributed software infrastructure from an optimized  
distributed topology, the built distributed software infrastructure configured to stream  
data to the client device from the source device without rendering the data by the  
source device, wherein at least one of the [[said]] software components is executable on  
each of:

the source device; and  
the client device[[. ]].

instantiating one or more child DMS; and  
delegating control of one or more of the plurality of software components to the  
one or more child DMS.

47. (Currently Amended) A system<sub>1</sub> comprising:

a source device which includes a software component that, when executed by the source device, streams data that is to be rendered;

a client device which includes a software component that, when executed by the client device, renders a stream of data; and

a parent distributed mediasession ("DMS"), which when executed by either the source device or the client device, provides a federated mechanism for control of:

the software component that, when executed by the source device, streams data without rendering the data that is to be rendered; and

the software component that, when executed by the client device, renders a stream of data[.].

the parent DMS instantiating one or more child DMS and delegating control of  
the software component to the one or more child DMS.